(12) INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(19) World Intellectual Property Organization
International Bureau

(43) International Publication Date
6 June 2002 (06.06.2002)

PCT

(10) International Publication Number
WO 02/45319 A2

(51) International Patent Classification[7]: H04L

(21) International Application Number: PCT/US01/44602

(22) International Filing Date:
29 November 2001 (29.11.2001)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:

60/250,899	1 December 2000 (01.12.2000)	US
60/265,466	30 January 2001 (30.01.2001)	US
09/921,171	2 August 2001 (02.08.2001)	US
09/969,060	2 October 2001 (02.10.2001)	US

(71) Applicant: O2 MICRO INTERNATIONAL LIMITED [—/—]; P.O. Box 32331 SMB, Grand Cayman (KY).

(71) Applicants and
(72) Inventors: DU, Sterling [US/US]; 895 Llima Court, Palo Alto, CA 94123 (US). CHAN, Reginia [US/US]; 1548 Plateau Avenue, Los Altos, CA 94024 (US). DENSHAM, William, L. [US/US]; 15591 Linda Avenue, Los Gatos, CA 95032 (US). HER, Sheau, Chuen [US/US]; 2215 Cuesta Drive, Milpitas, CA 95035 (US).

(72) Inventors: LAM, James; 34760 Swain Common, Fremont, CA 94555 (US). IVCHENKO, Volodymyr; 1560 San Tomas Aquino Road, #5, San Jose, CA 95130 (US).

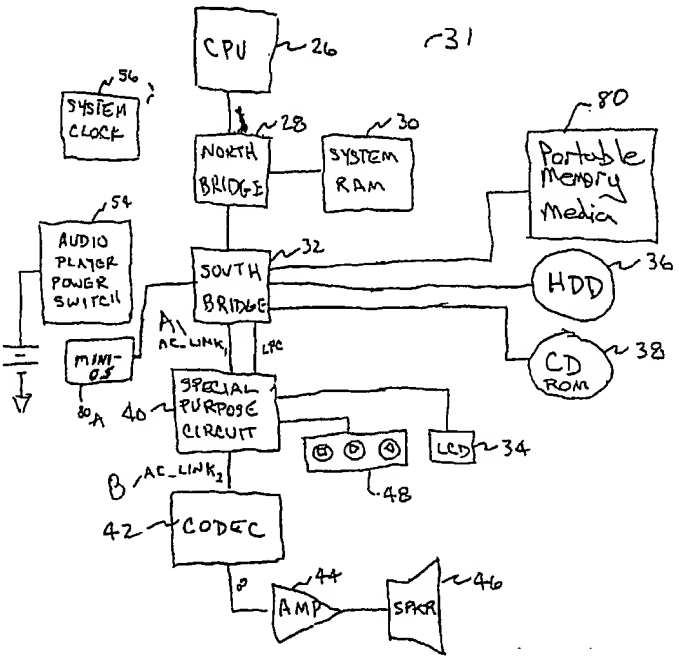
(74) Agents: PFLEGER, Edmund, P. et al.; Hayes, Soloway, Hennessey, Grossman & Hage, P.C., 130 W. Cushing Street, Tucson, AZ 85701 (US).

(81) Designated States *(national)*: AE, AG, AL, AM, AT, AU, AZ, BA, BB, BG, BR, BY, BZ, CA, CH, CN, CO, CR, CU, CZ, DE, DK, DM, DZ, EC, EE, ES, FI, GB, GD, GE, GH, GM, HR, HU, ID, IL, IN, IS, JP, KE, KG, KP, KR, KZ, LC, LK, LR, LS, LT, LU, LV, MA, MD, MG, MK, MN, MW, MX, MZ, NO, NZ, PH, PL, PT, RO, RU, SD, SE, SG, SI, SK, SL, TJ, TM, TR, TT, TZ, UA, UG, UZ, VN, YU, ZA, ZW.

(84) Designated States *(regional)*: ARIPO patent (GH, GM, KE, LS, MW, MZ, SD, SL, SZ, TZ, UG, ZM, ZW), Eurasian patent (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European patent (AT, BE, CH, CY, DE, DK, ES, FI, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE, TR), OAPI patent

[Continued on next page]

(54) Title: LOW POWER DIGITAL AUDIO DECODING/PLAYING SYSTEM FOR COMPUTING DEVICES



(57) Abstract: A low-power digital audio decoding and playing system and method for computing devices provides a low-cost, low power-consumption, long-battery-life audio playing and decoding system, which may be used to play compressed audio files of various formats. In one aspect, a computer system adapted to play audio files comprises a system CPU, memory, at least one drive comprising compressed audio data, and an audio controller coupled to the system CPU, memory and drive. The audio controller is adapted to cause the drive to read the compressed audio data from the drive, to cause the system CPU to decompress the compressed audio data from the drive into decompressed audio data, to cause the decompressed audio to be stored in the memory, and to cause the decompressed audio data to be retrieved from the memory for playing.

WO 02/45319 A2

(BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, ML, MR, NE, SN, TD, TG).

Published:

— *without international search report and to be republished upon receipt of that report*

For two-letter codes and other abbreviations, refer to the "Guidance Notes on Codes and Abbreviations" appearing at the beginning of each regular issue of the PCT Gazette.

LOW POWER DIGITAL AUDIO DECODING/PLAYING SYSTEM FOR COMPUTING DEVICES

This application is a continuation-in-part of U.S. patent application no. 09/921,171 filed August 2, 2001 entitled "Low Power Digital Audio Decoding/Playing System for Computing Devices", which claims the benefit of provisional application serial No. 60/250,899, filed on December 1, 2000, entitled "Low Power Digital Audio Decoding System for Computing Devices" and provisional application serial No. 60/265,466, filed on January 30, 2001, entitled "Low Power Digital Audio Decoding/Play System for Computing Machines".

BACKGROUND OF THE INVENTION

1. Field of the Invention

The present invention relates generally to portable devices (e.g., notebook computers) for reproducing audio recordings, and more particularly, to low-power hardware and software for decoding and reproducing compressed audio recordings in a variety of compression formats from a variety of sources. While particular utility for the present application is in the reproduction of MP3 digital audio files, especially for use with portable computers, other utilities are contemplated herein.

2. Description of Related Art

Presently there exist various portable devices for replaying digital audio recordings that have been compressed in accordance with one or more compressed audio digital recording formats, e.g., MPEG (Moving Picture Experts Group) Audio Layer-3 (MP3), Windows® Media Audio (WMA), and Advanced Audio Coding (AAC). To date, the most popular format has been MP3, a compression scheme that results in about a 10:1 compression of the size of digital music files. These devices can be divided into two classes, those which store the compressed digital audio recordings in an electronic solid-state memory, and those which record the compressed digital audio for subsequent reproduction using an electro-mechanical device such as a compact disk ("CD") player or on a hard disk drive of a digital computer.

For example, portable devices for playing MP3 compressed digital audio

recordings that use electronic solid-state memory, e.g., flash-memory, are capable of storing about ten (10) music selections. With an add-in memory card, such devices can carry a total of about twenty (20) music selections. These MP3 players that store the MP3 compressed digital audio recordings in an electronic solid-state memory consume comparatively little electrical power. Thus, such MP3 players provide an extended playing interval without having to power the computer's CD-ROM or hard disk drive.

U.S. Patent No. 6,226,237, entitled "Low Power CD-ROM Player for Portable Computers", issued May 1, 2001 (the "'237" patent), which is hereby incorporated by reference in its entirety, describes how a conventional notebook computer, when simply playing a conventional music CD, consumes an unnecessarily large amount of electrical energy. That is largely due to the large number of background functions that are unrelated to the playing of music that the Operating System (e.g., Windows®) is performing whenever the computer is turned on. That excessive electrical energy consumption for functions unrelated to the function the user is performing at the moment, i.e., playing music, quickly drains the battery of a notebook computer of power that could more prudently be applied at another time in performance of microprocessor intensive tasks such as word processing and spreadsheet analysis. The solution presented in the '237 patent is a state machine that operates when main power to the portable device is OFF. The invention of the '237 patent couples a CD-ROM to the audio subsystem (when main power is OFF) so that CDs can be played, without excessive battery drain, or without having to boot up the portable computer.

The prior art also includes silicon solutions that are dedicated function integrated circuits (ICs) or incorporated into application-specific integrated circuits, or ASICs. These are usually expensive solutions as the digital signal processor (DSP) required in a dedicated chip results in a large, costly integrated circuit. One of the results is the use of a larger amount of PCB (printed circuit board) space.

Further, the 15 to 20 MIPS (million instructions per second) decode engine known in the art must be continuously running to generate the audio stream for the Codec. Additionally, the dedicated decode engine needs to have the high-power-consuming hard disk drive (HDD) continuously operating. These approaches are

limited to functioning only with MP3 compression, thereby eliminating the opportunity to adapt the system to newly emerging music compression algorithms, such as Microsoft's WMA or the music industry's proposed Secure Digital Music Initiative (SDMI) for secure audio.

Dedicated silicon solutions known in the art employ a DSP that must constantly be decoding the compressed audio files from a hard disk drive, which must therefore be constantly reading the audio files. Such known methods require much power, resulting in a fast battery discharge, (e.g., much faster than the possible 4 to 10 hours of desired use on a transoceanic flight).

Thus, known hardware MP3 decoder and players requiring an IC implementation and a hard disk drive being accessed non-stop are high in power consumption, difficult to upgrade, and expensive.

The present invention provides a solution that is low in power consumption, can be upgraded in the field for various music compression formats, is expected to cost no more than half the cost of the currently available hardware implementation, and may be made capable of playing up to hundreds of musical selections, while only having to access the HDD or CD-ROM less than 0.5% of the time.

SUMMARY OF THE INVENTION

It is becoming more and more desirable for mobile platform companies to add MP3 and other compressed audio player capability to their products, with low cost, while providing very long music playing time, and perhaps even a player that can be later upgraded to other compression formats by the owner. These mobile platform companies may also want to market differentiate their products within a very short development time frame.

The music playing solution of the present invention utilizes a special purpose circuit in combination with the mini-OS (operating system) software of the present invention. The present invention uses the embedded computing power of the standard CPU to perform the file decompression. Since today's CPUs with clock rates of 500 MHz to 1 GHz have at least an order of magnitude higher processing power than the real time DSP engines used in currently available MP3 player/decoders, these powerful CPU processors can often finish the decoding process in less than 10% of

the available time. The CPU may then be set to idle by the present invention for more than 90% of the time, saving large amounts of power and thus greatly slowing the discharge of the battery and extending the useful time of the equipment under battery power on a single charge.

The present invention is unlike the real-time DSP engines known in the art, which require a constant data stream from the HDD, and which result in high power consumption, since the HDD is being accessed all the time. Using the technology of the present invention, the HDD may be accessed less than 0.5% of the time with a typical complement of memory, i.e., 128MB RAM. This results in a dramatic reduction in the rate at which power is dissipated from the equipment battery. Further, minimal PCB changes are required for the present invention, thus resulting in the quick adoption of new product features in PCs.

There are many possible music compression algorithms. Compression algorithms other than MP3 include WMA, AAC, and the proposed SDMI. The software decompression methodology of the present invention can be easily modified to decode any compression scheme, or with a software installation process, all the various compression schemes. This flexibility allows the adaptation to new and different algorithms, as they become popular, by permitting an after-market upgrade of computers equipped with the present invention. Also, since this portion of the present invention is a software system, new updates and/or algorithms may be downloaded (e.g., from the Internet) to upgrade machines in the field, eliminating the necessity for consumers to buy multiple players/decoders in order to listen to audio files having different compression formats.

Thus, the present invention provides a low-cost, low power-consumption, long-battery-life audio playing and decoding system, which may be used to play audio files of various formats.

In one aspect, a computer system adapted to play audio files comprises a system CPU, memory; at least one drive comprising compressed audio data residing in one or more audio files, a play list software program for selecting and storing a play list comprising one or more of the audio files, a first operating system adapted to control at least the system CPU and memory, and a second operating system stored in

BIOS and adapted to retrieve the play list and cause the drive to read at least one audio file of the play list, to cause the system CPU to decompress the compressed audio data of the file and provide decompressed audio data, and to cause the decompressed audio data to be stored in memory.

In another aspect, a computer system adapted to play audio files comprises a drive comprising at least one audio file, an audio controller, and an operating system stored in BIOS, the operating system controlling the audio controller, so as to cause the audio controller to play at least one audio file.

In a further aspect, a computer system adapted to play audio files comprises: compressed audio data, a system CPU, an audio controller, a first operating system adapted to control at least the system CPU, a second operating system controlling the audio controller and system CPU, so as to cause the system CPU to decompress the compressed audio data, and a switch, the activation of the switch causing the second operating system to boot.

In yet another aspect, a computer system adapted to play audio files comprises a system CPU, memory, at least one drive comprising compressed audio data residing in one or more audio files, a play list software program for selecting a play list comprising one or more of the audio files, and an audio controller coupled to the system CPU, memory and drive. The audio controller is adapted to cause the drive to read at least one audio file of the play list, to cause the system CPU to decompress the compressed audio data of the file and thereby provide decompressed audio data, and to cause the decompressed audio data to be stored in memory.

In process form, a method of playing audio files on a computer system comprises: booting a first operating system; creating and storing a play list comprising a list of compressed audio files residing on one or more drives of a computer system having at least a drive, a CPU, and a memory; terminating the first operating system; booting a second operating system upon activation by a switch; reading the play list; reading the compressed audio files from the drive based on the play list; providing the compressed audio data to the CPU for decompressing the data of the compressed audio file into decompressed audio data; storing the decompressed audio data in memory; and retrieving the decompressed audio data from the memory

for playing.

In another process form, a method of playing audio files on a computer system comprises: reading compressed audio data from the drive of a computer system having at least a drive, a CPU, and a memory; providing the compressed audio data to the CPU for decompressing the compressed audio data, thereby providing decompressed audio data; and storing the decompressed audio data in memory.

BRIEF DESCRIPTION OF THE DRAWINGS

Figure 1 is a block diagram representation an exemplary operational flow of one embodiment of the present invention;

Figure 2 is a flow diagram of an exemplary power up of the mini-OS and initiation of the player function, in one embodiment of the present invention;

Figure 3 is a block diagram of an exemplary audio player system consistent with one embodiment of the present invention;

Figure 4 is a block diagram of the internal portion of an exemplary special purpose circuit, in relation to the other components that interface with it, in one embodiment of the present invention; and

Figure 5 is another block diagram of an exemplary audio player system consistent with another embodiment of the present invention.

DETAILED DESCRIPTION OF EXEMPLARY EMBODIMENTS

The present invention comprises mini-OS (operating system) software and a hardware interface between the South Bridge and Codec to play the musical selections (or other stored audio) desired by the user. The mini-OS software of the present invention performs only those functions and enables those elements of the portable computer that are needed, when they are needed, to play the selected music, without performing all of the background functions performed by the full system operating system, e.g., Windows®, and without accessing the monitor circuitry and monitor screen of the portable computer. Additionally, the mini-OS of the present invention only accesses the HDD when compressed files are being transferred to RAM. Thus, it will be seen that the mini-OS software portion of the present invention performs both power saving and file management functions when playing audio.

Figure 1 is a block diagram representation of the operational flow of the

exemplary software compressed audio player in one embodiment of the present invention.

The operational concept illustrated in Figure 1 is as follows:

1st: A browser, running on a full system operating system, e.g., Windows®, of the portable computer is initially used to download compressed music files (for example 1000 songs) onto the PC hard disk drive (HDD) (2) (e.g., using 4 gigabytes of HDD space) at some time prior to the time at which the user desires to use the portable computer as an audio player and a playlist is created, comprising the songs the user desires to hear at a later time;

2nd: When the user desires to use the portable computer as an audio player, once the desired music files are on the HDD, the user operates an audio player on-switch to turn the portable computer fully on, boot up the entire computer, load in the mini-OS of the present invention instead of the usual Microsoft Windows® OS (the full system operating system is not opened) with the power saving initialization subroutines and initializes only those portions of the portable computer as necessary, and the file management subroutines initialize the song play list or book generated in step 1, of a substantial number of songs, for desired music listening under direction of the user;

3rd: The mini-OS software is then copied from the HDD (2) to RAM (4), and then the first set of compressed files from the song play list is copied from the HDD (2) to the system RAM (4) also using the mini-OS software of the present invention. For example, in today's PC's 128 Mbytes is a typical system RAM size, with the mini-OS software of the present invention taking about 8 Mbytes of the RAM, leaving approximately 120 Mbytes for use as a compressed music memory (i.e., a cache or buffer, using system memory, dedicated memory, or other memory). That 120 Mbytes represents about 2 hours of continuous compressed music with a compression ration of 10:1, typical of MP3 files. Similarly, in the case when flash media is used

for MP3 storage, all or most of the contents of the flash media card can be copied to the system RAM (4), thus minimizing the access of the flash media reader and allowing for a more responsive control over the MP3 files;

4th: The file management software of the present invention sequentially delivers portions of the first music file to the CPU (6) where the decode algorithm decompresses each file using the file management software of the present invention stored in RAM (4). Once decoded, the PCM audio data is transferred in one of three ways: the CPU delivers the PCM audio data to the South Bridge (see Figure 3 (32)) FIFO buffer; the DMA in the South Bridge transfers the data internally within the South Bridge to the FIFO buffer; or the special purpose circuit transfers the data to the FIFO buffer from the LPC interface. The FIFO buffer then sequentially feeds each piece of decoded music to Codec (8) (also see Figure 3 (42)), through the special purpose circuit of the present invention, where the decoded signal is converted from digital to analog. Then the output signal from the Codec (8) is amplified (10) (also see Figure 3 (44)) to drive the speakers and/or headset (see Figure 3 (46)).

5th: While the final song of the first set from the play list is playing from memory, the file management software of the present invention stored in the RAM (4, 30) returns control to the 4th step to retrieve the next set of compressed music files from the memory of the RAM, as determined by the earlier scripted song play list developed in the 1st step. Thus, the 4th and 5th steps are repeated for each set of compressed music files until the last music selection in the set plays. At that point in time control returns to the 3rd step to load another set from the play list, which is similarly played through the 4th and 5th steps. When the last song is played from the overall play list of the 2nd step, or when the user turns off the music player function, the operation of the player ceases.

The mini-OS power saving software of the present invention ensures that the CPU, Peripheral Chips, HDD and other controllable system elements will be in idle state for the highest percentage time possible. An interesting attribute of the solution offered by the present invention is that the higher the MIPS (Million Instructions Per Second) capacity of the CPU, the smaller percentage of time the CPU will spend performing the decode function. This means that higher performance CPU's will demonstrate even lower power usage when playing compressed music performances, thus saving even more battery power and further extending the length of time that the battery maintains sufficient charge to power the portable computer.

The mini-OS monitors the audio control buttons (e.g., play, fast forward, rewind, pause, scan, previous track, next track, first track, last track, fast forward/rewind while listening, audio source/media select (e.g., HDD or CD), etc.) (see Figure 3 (48)) for user actuation through the special purpose circuit (see Figure 3 (40)) of the present invention, and communicates user requests to the mini-OS file management software of the present invention. Optionally, a small LCD display (see Figure 3 (34)) can be connected to the special purpose circuit to provide visual status indicators (e.g., Song #, Song titles, track #, Playtime & icons) under control of the mini-OS display management subroutines.

The mini-OS power saving software of the present invention primarily manages the usage of the CPU, and the MP3 storage devices such as CD, HDD, and flash media such as SD (Secure Digital) cards, MMC (Multimedia Card), memory stick, and SMC (Smart Media Card), while maintaining the rest of the system, including the memory, corelogic chipsets, in a fully on and functional state. Secondary power saving is applied to other PC subsystems to minimize power usage still further by putting them in an idle state.

For example, with a 500 MHz Pentium III CPU having about 225 MIPS of processing power and the decode algorithm requiring about 15 MIPS, the CPU will be operating less than 10% of the time. The other 90-95% of the time the CPU will be in a standby mode that requires only milliamps of current. Alternatively, the CPU can be run at a slower clock speed, which is usually an option provided by most of today CPUs, such as the AMD's Athlon CPU. Similarly the HDD is accessed during the

time it takes to fill or refill the RAM. Thus, since the average song takes about 4 minutes to play and the RAM holds about 30 songs for 120 Mbytes, and since the HDD needs 1-5 seconds to spin up and only several seconds to load the song play list into RAM, the total access time for the HDD may be 30 seconds out of 120 minutes of play time; a ratio of 1:240, less than 0.5% of full power operating time. These factors add to the power savings gained by using the mini-OS of the present invention instead of the full operating system of the portable computer. The result of the overall power consumption of the present invention is very low when the portable computer is in the music play mode, and that directly translates into the battery maintaining a useful charge level for a much longer time than allowed by the prior art. As those skilled in the art will recognize, the compressed music data of this invention may reside on a hard disk, on other magnetic (e.g., tape) media, optical (e.g., CD-ROM) media, flash media (e.g., SD cards, MMC, memory stick, SMC), or any other storage medium.

Figure 3 is a generalized overall block diagram of an exemplary system 31 consistent with one embodiment of the present invention. The majority of the blocks in system 31 are components known in the art and are generally included in all PC computers for producing sound through the speaker of the computer. Shown here is a system clock 56, which, for simplicity of Figure 3, is not shown connected to the various components that need a clock signal. Additionally, CPU 26 is shown interfacing with North Bridge 28. In turn, North Bridge 28 interfaces with system RAM 30 and South Bridge 32. Then South Bridge 32 interfaces with HDD 36 and CD-ROM 38. Typically South Bridge 32 also interfaces directly with Codec 42 through AC_link; however, in the exemplary system 31 shown, special purpose circuit 40 (see discussion of Figure 4 below) is inserted between South Bridge 32 and Codec 42 to enable the playing of compressed digital audio in conjunction with the mini-OS 80 of the present invention from system RAM 30, without affecting the ability to play non-compressed analog audio. In this configuration, the mini-OS 80 is stored in the BIOS, although those skilled in the art will recognize that the mini-OS could alternatively be stored in its own ROM (either within special purpose circuit 40 or external to it), a hard disk, or other media. Thus, AC_link_1 from South Bridge 32

is coupled to special purpose circuit 40, which performs the decompression function as necessary, and then provides any audio signals to Codec 42 via AC_link_2. Codec 42 then performs the usual function on all signals received from special purpose circuit 40 and applies the audio signals to amplifier 44, to be played on speaker 46 or headphones (not shown). In system 31, AC_link_1 looks and behaves like the standard AC_link to South Bridge 32, and AC_link_2 looks and behaves like the standard AC_link to Codec 42, making it appear to those portions of the computer that audio functions are being performed as during normal (i.e., known in the art) audio play, thus having minimal or no impact on the operation of South Bridge 32 and Codec 42. Also shown in Figure 3 are function switches 48, small LCD display 34 and audio player power switch 54, which function as described hereinbelow with reference to Figure 4.

Figure 4 includes a detailed block diagram of the internals of special purpose circuit 40 and related details of the other portions of the computer that the special purpose circuit interfaces without showing all of the details of the rest of the computer system. Special purpose circuit 40 may be produced as an IC to minimize the PCB space needed to incorporate embodiments of the present invention into portable computers. South Bridge 32 is shown with the standard AC 97 controller 50 and LPC (low pin count) controller 52 to the left of special purpose circuit 40 with the standard bidirectional links AC_link_1 and LPC Bus between them, and the unidirectional IRQ (Interrupt Request) link from special purpose circuit 40 to South Bridge 32. To the right, special purpose circuit 40 provides uncompressed audio to AC 97 Codec 42 via AC_link_2. Also, to the right, function keys 48, and below LCD 34, are each shown connected to special purpose circuit 40. Additionally, Figure 4 includes system clock 56 connected to various components, and in the lower left, audio player power switch 54. Power switch 54 is provided so that when the user initiates the player mode via power switch 54, only the mini-OS (instead of the full system OS) is initiated, for use in a system consistent with the present invention.

Internal to special purpose circuit 40 are switches 60 that interface with both AC_link_1 and AC_link_2 and function in response to settings in an internal register of register block 66, with switches 60 closed connecting $AC\text{-}link_1$ with AC_link_2 when

the PC functions normally with the full system OS, and with switches 60 open when a system consistent with the present invention is employed. The LPC path is coupled to LPC interface. Switches 60 and AC_link_2 are coupled to state machine 64, while another port of state machine 64 is coupled, via bus 74, to the output of LPC interface 62, as well as register block 66, function key interface 68 and LCD interface 72. A second port of register block 66 is also coupled to a third port of state machine 64. Function keys 48 are coupled to function key interface 68, and LCD 34 is coupled to LCD interface 72. Also, function key interface 68 provides a signal to register block 66 when one of the function keys 48 is selected by the user. Audio player power switch 54, which is operated by the user in the second step discussed above, may be used to activate the PC to operate as described hereinabove. Switch 54 is shown connected to the DC voltage source of the portable computer and not to any particular block in Figure 4, since that connection varies depending on several factors controlled by the manufacturer of the computer on which an embodiment of the present invention is installed.

More specifically, the blocks within special purpose circuit 40 operate as follows:

LPC Interface

Special purpose circuit 40 includes LPC (Low Pin Count) interface 62 to interface with LPC controller 52 in South Bridge 32.

The LPC interface 62 is used to by CPU 26 to:

(1) read the function key input registers in register block 66;

(2) set the control register in register block 66 to control the AC97 Codec 42;

(3) get the audio PCM (Pulse Code Modulation) data from the system memory (RAM 30); and

(4) perform clock throttling control.

The setting in the mode register of register block 66 controls the state of switches 60 to switch the special purpose circuit 40 between the normal computer operation mode with switches 60 closed (e.g., running Microsoft Windows® OS) and the mode of a system consistent with the present invention, with switches 60 open

(running the mini-OS) to play compressed audio files.

South Bridge AC97 Controller 50 interface (AC_Link$_1$ from host)

During the normal computer operation mode, switches 60 are closed with the South Bridge AC97 Controller 50 interface connected directly through, closed switches 60, to AC97 Codec 42 to generate audio output as if special purpose circuit 40 were not present. To play compressed audio files, switches 60 are open when the mini-OS is running, and state machine 64 controls AC97 Codec 42.

AC97 Codec interface (AC_Link$_2$ to AC97 Codec 42)

When the computer is running under control of the mini-OS, switches 60 are open. State machine 64 then controls the AC_link$_2$ in response to the settings of the register block 66 set by the host (CPU 26) to generate the controls for AC97 Codec 42 (e.g., switching the sampling frequency, controlling volume, sending the PCM data to the Codec 42, setting the Codec 42 to the power saving mode or waking Codec 42 from the power saving mode).

Function Key Input Interface 68

Function key interface 68 receives the user selections from function keys 48 and stores the selections in internal registers to be read by CPU 26.

LCD interface 72

LCD interface 72 is only necessary if LCD 34 is used to provide status information to the user. The purpose, when used, is to show player status on low cost LCD 34 when the system consistent with the present invention is used. Status of the audio track number of the selection playing, status icons (e.g., Play) and other generic status icons may be programmed into the system and displayed for any other purpose.

Operation Modes

(A) Normal Operation Mode:

When the PC is fully powered and running under the full system OS, the various functions of special purpose circuit 40 are bypassed and switches 60 are closed, as discussed above. In the normal mode, the computer system uses the South Bridge AC97 Controller 50 to directly control the AC97 Codec 42 through the AC_link (in the Normal mode AC_link$_1$ and AC_link$_2$ are the same since switches 60 are closed. The special purpose circuit does not intercept of modify the AC_link

signals.

(B) Compressed Audio Performance Mode:

When switch 54 has been closed, the system runs under the control of mini-OS, and special purpose circuit 40 is empowered and runs in the compressed audio performance mode. The South Bridge AC97 Controller 50 is isolated from the AC97 Codec 42 in this mode since switches 60 are open.

In the compressed audio performance mode, the host (CPU 26) sets the internal registers of register block 66 to control the data flow to the AC97 Codec 42, and to perform the various power management functions.

A Power Saving Control Method in Compressed Audio Performance Mode

A flexible control method of the special purpose circuit 40 is provided to minimize the system control cycles and power consumption in the performance mode. The system memory (RAM 30) is used to pass most of the control commands to the special purpose circuit 40, instead of CPU 26, which minimizes the time that CPU 26 needs to access high speed external bus other than a standby level. This considerably reduces the power load on the portable computer battery in this mode.

CPU 26 also sets the system control memory registers in register block 66. State machine 64 bases operation on those register settings to obtain control words and PCM data automatically through the LPC interface 62. The control words in the system memory (RAM 30) are fetched into the internal registers, and the state machine 64 decodes the control words to determine if PCM or audio data is ready. If the audio data is ready, the state machine 64 continues to fetch the audio data and send it to the AC97 Codec 42. The control words in the system memory (RAM 30) can also be used to indicate the sampling frequency of the PCM data. So, the state machine 64 can set AC97 Codec 42 to the appropriate frequency before the PCM data is sent.

Those skilled in the art will recognize that a headphone or headset system may comprise further functionality than described hereinabove, e.g., a volume control, or the audio control buttons may be integrated thereto.

It should also be recognized that a special purpose circuit consistent with the invention may be integrated into a full-time compressed (and/or non-compressed)

audio playing system capable of playing music regardless of the operation of the rest of the system. In this configuration, the special purpose circuit and mini-OS are provided, as well as a software driver for handling interrupts from the function buttons under Windows®. In this configuration, when the rest of the system is either fully on (S0) or in "sleep" (suspend to RAM or S3) mode, the system may be configured to begin execution of a custom or standard audio player, e.g., Music Match or Windows® Media Player, running under Windows®, which may be adapted to play the compressed audio files stored in the play list. In this scenario, the function buttons may be adapted for use in a passthrough-type mode using the accompanying software driver to control various features of the audio player software, e.g., Music Match, instead of controlling the special purpose circuit. When the primary operating system such as Windows® is either fully off (S5) or in "hibernate" (suspend to HDD or S4) mode, operation of the special purpose circuit may proceed to play compressed audio files from the play list as described hereinabove, wherein the function buttons control the special purpose circuit.

It is noted that the power states described above (i.e., fully on, sleep/suspend to RAM, fully off, hibernate/suspend to HDD) are often referred to using the Advanced Configuration and Power Interface ("ACPI") standard conventions, as follows: The typical operating system (e.g., Windows®) supports six system power states, referred to as S0 (fully on and operational) through S5 (power off). Each state is characterized by the following: power consumption, i.e. how much power the computer uses; software resumption, i.e, from what point the operating system restarts; hardware latency, i.e., how long it takes to return the computer to the working state; and system context, i.e. how much system context is retained, or whether the operating system must reboot to return to the working state. State S0 is the working state. States S1, S2, S3, and S4 are sleeping states, in which the computer appears off because of reduced power consumption but retains enough context to return to the working state without restarting the operating system. State S5 is the shutdown or off state. A system is waking when it is in transition from the shutdown state (S5) or any sleeping state (S1-S4) to the working state (S0), and it is going to sleep when it is in transition from the working state to any sleep state or the shutdown

state. the system cannot enter one sleep state directly from another; it must always enter the working state before entering any sleep state. For example, a system cannot transition from state S2 to S4, nor from state S4 to S2. It must first return to S0, from which it can enter the next sleep state. Because a system in an intermediate sleep state has already lost some operating context, it must return to the working state to restore that context before it can make an additional state transition.

Referring now to Figure 2, in conjunction with Figure 3, an exemplary sequence 200 for the power up of the mini-OS and initiation of the player function, in one embodiment of the present invention, is illustrated. As stated above, at some time prior to the initiation of the audio player function of a PC equipped with the present invention, the user downloads (not shown in Figure 2) the audio files of interest to the HDD 36 or burns a CD-ROM that is placed in the CD-ROM drive 38 for use with the audio player feature of the present invention. As shown, at step 201, the sequence 200 begins when the user presses either an audio player power switch 54 or the computer's main power switch (not shown in Figure 3), to turn the system on. A determination is then made, at step 202, whether the computer is to boot in normal operation mode or compressed audio performance mode. This determination is typically made in the BIOS, based on whether the computer's power switch or an audio player power switch 54 was used to turn on the computer, although those skilled in the art will recognize that this determination could alternatively be made by an application program or an operating system that provides such capability (e.g. Windows® 98). If the computer's power switch was used to turn on the computer, then the system boots to normal operation mode, at step 203, and the normal operating system (e.g., Windows® 98) is loaded into system RAM 30 and executed. If an audio player power switch 54 was used to turn on the computer, the mini-OS is loaded into system RAM 30, at step 204. At step 203, the mini-OS initializes the system components including one or more of the North Bridge 28, South Bridge 32, special purpose circuit 40, hard drive 36, CD-ROM drive 38, codec 42, and CPU 26.

Since no audio decompression request will be pending upon system initialization (i.e., the memory buffer is not full), which determination is made at step 208, the system waits for input from one of the function keys 48, at step 207, until one

of the function keys 48 is pressed, at which point the appropriate function is executed and the LCD display updated, as appropriate, at step 206. If the command includes a request from the user to play audio, an audio decompression request will be pending at this time, which determination is made at step 208. Since no compressed audio file(s) are in system memory 30 upon the initial request to play audio, which determination is made at step 209, the compressed audio file(s) are read from the HDD 36 and/or CD-ROM drive 38 and loaded into system memory 30, at step 210. After the compressed audio files are loaded into system memory at step 210, or if the audio file(s) are already in system memory, which determination is made at step 209, the audio files are then decompressed, at step 211, using the system CPU 26. DMA transfer(s) to the codec 42 are initialized for the decompressed audio data, at step 212, and then the output signal from the Codec 42 is amplified (not shown in Figure 2) by the amplifier 44 to drive the speakers and/or headset 46. After the DMA transfer(s) are initialized, at step 212, control loops back to step 208, to determine whether an audio decompression request is pending.

Playlist Software Operation

Figure 5 is another generalized overall block diagram of an exemplary system 31 consistent with another embodiment of the present invention. In this exemplary embodiment, the system 31 includes portable memory media 80 that can be used to hold the playlist data and/or compressed file data. The memory media 80 can be SmartCard media, Memory Stick media, PCMCIA memory media and/or other portable media known in the art. If the system is ON and media is detected as being present at the portable memory media location (e.g., by insertion of a Smart Card, PCMCIA, CardBus card, Memory Stick or other media into an appropriate slot), the memory reader generates an interrupt to the South Bridge 32. The special purpose circuit 40 of this embodiment also receives the interrupt and generates a command to tell the operating system to launch an appropriate application (e.g., Windows Media Player) to read the playlist data on the memory device 80. In this instance, the application takes control to read the playlist file and retrieve the audio data, either from the memory device 80 or some other location specified in the playlist file. Similarly, when the mini-OS is operational, the special purpose circuit 40 is adapted

to check if a memory device 80 is present, and to scan the device for playlist data. The system then operates as described above.

The playlist file, as described herein, is a generalized data file that is constructed by a user having a desired MP3 song sequence. The playlist file also includes disk path information to instruct the application as to where to locate the desired MP3 data. Certain operating systems permit users to change drive letters on-the-fly. Accordingly, the playlist software reads the volume serial number (VSN) given by the operating system to a particular drive. The serial number does not change (unless intentionally changed by reformatting the drive), and thus, the playlist software can track the playlist data regardless if the user reassigns a particular drive letter. This feature also works similarly with switchable devices such as disk drives.

It should be recognized by those skilled in the art that, although the above-described embodiments utilize a hardware-based OS selection (i.e., pressing main power button boots to Windows®, while pressing audio control button boots to mini-OS), other OS selection methods are contemplated, as well. Such selection methods include, e.g., using a batch file or other scripting or software-based method to shut down a first OS and boot to the second OS. Those skilled in the art will also recognize that the mini-OS of the present invention could conceivably be implemented as part of a larger OS (e.g., a GUI-based OS, such as Windows®, LINUX, etc.) or as a software component named something other than an "operating system", (e.g., a "driver", an "algorithm", a "script", "code", a "program", an "executable", a "routine", a "subroutine", a "utility", etc.), instead of being implemented as an entirely separate operating system. Such embodiments are contemplated to be within the scope of the present invention.

Although the present invention has been described in terms of the exemplary embodiments provided herein, it is to be understood that such disclosure is purely illustrative and is not to be interpreted as limiting. Consequently, without departing from the spirit and scope of the invention, various alterations, modifications, and/or alternative applications of the invention will, no doubt, be suggested to those skilled in the art after having read the preceding disclosure. Accordingly, it is intended that the following claims be interpreted as encompassing all alterations, modifications, or

alternative applications as fall within the true spirit and scope of the invention.

What is claimed is:

1. A computer system adapted to play audio files, said computer system comprising:

a system CPU;

memory;

at least one drive comprising compressed audio data, said compressed audio data residing in one or more audio files;

a play list software program for selecting and storing a play list comprising one or more of said audio files;

a first operating system adapted to control at least said system CPU and said memory; and

a second operating system, said second operating system being stored in BIOS and adapted to retrieve said play list and cause said drive to read at least one said audio file of said play list, to cause said system CPU to decompress the compressed audio data of said file and provide decompressed audio data, and to cause said decompressed audio data to be stored in said memory.

2. A computer system adapted to play audio files, said computer system comprising:

a system CPU;

memory;

at least one drive comprising compressed audio data;

a first operating system adapted to control at least said system CPU and said memory; and

a second operating system, said second operating system being stored in BIOS and adapted to cause said system CPU to decompress said compressed audio data and store said decompressed audio data in said memory.

3. A computer system adapted to play audio files, said computer system comprising:

a system CPU;

memory;

at least one drive comprising compressed audio data; and

an operating system stored in BIOS, said operating system being adapted to cause said system CPU to decompress said compressed audio data and store said compressed audio data in said memory.

4. A computer system adapted to play audio files, said computer system comprising:

compressed audio data;

a system CPU; and

an operating system stored in BIOS, said operating system being adapted to cause said system CPU to decompress said compressed audio data.

5. A computer system adapted to play audio files, said computer system comprising:

compressed audio data;

a system CPU;

an audio controller; and

an operating system stored in BIOS, said operating system controlling said audio controller and said system CPU, so as to cause said system CPU to decompress said compressed audio data.

6. A computer system adapted to play audio files, said computer system comprising:

compressed audio data;

a system CPU;

an audio controller;

a first operating system adapted to control at least said system CPU; and

a second operating system stored in BIOS, said second operating system controlling said audio controller and said system CPU, so as to cause said system CPU to decompress said compressed audio data.

7. A computer system adapted to play audio files, said computer system comprising:

a system CPU;

memory;

at least one drive comprising compressed audio data, said compressed audio data residing in one or more audio files;

a first operating system adapted to control at least said system CPU and said memory;

a play list software program executable under said first operating system, said play list software program being adapted to permit selection and storage of a play list comprising one or more of said audio files; and

a second operating system, said second operating system being stored in BIOS and adapted to retrieve said play list and cause said drive to read at least one said audio file of said play list, to cause said system CPU to decompress the compressed audio data of said file and provide decompressed audio data, and to cause said decompressed audio data to be stored in said memory.

8. A computer system adapted to play audio files, said computer system comprising:

at least one drive comprising audio data, said audio data residing in one or more audio files;

a system CPU;

memory;

a first operating system adapted to control at least said system CPU and said memory;

a play list software program executable under said first operating system, said play list software program being adapted to permit selection and storage of a play list comprising one or more of said audio files; and

a second operating system, said second operating system being stored in BIOS and adapted to retrieve said play list and cause said drive to read at least one said audio file of said play list, and to play said at least one said audio file of said play list.

9. A computer system adapted to play audio files, said computer system comprising:

a drive comprising at least one audio file;

an audio controller; and

an operating system stored in BIOS, said operating system controlling said audio controller, so as to cause said audio controller to play said at least one audio file.

10. A computer system adapted to play audio files, said computer system comprising:

a system CPU;

a drive comprising at least one compressed audio file; and

an operating system stored in BIOS, said operating system controlling said system CPU, so as to cause said system CPU to decompress said at least one audio file.

11. A method of playing audio files on a computer system, said method comprising:

booting a first operating system;

creating and storing a play list comprising a list of compressed audio files residing on one or more drives of a computer system having at least a drive, a CPU, and a memory;

terminating said first operating system;

booting a second operating system upon activation by a switch;

reading said play list;

reading said compressed audio files from said drive based on said play list;

providing said compressed audio data to said CPU for decompressing the data of said compressed audio file into decompressed audio data;

storing said decompressed audio data in said memory; and

retrieving said decompressed audio data from said memory for playing.

12. A method of playing audio files on a computer system, said method comprising:

booting a first operating system;

creating and storing a play list comprising a list of compressed audio files residing on one or more drives of a computer system having at least a drive, a CPU, and a memory;

terminating said first operating system;

booting a second operating system;

reading said play list;

reading said compressed audio files from said drive based on said play list;

providing said compressed audio data to said CPU for decompressing the data of said compressed audio file into decompressed audio data;

storing said decompressed audio data in said memory; and

retrieving said decompressed audio data from said memory for playing.

13. A computer system adapted to play audio files, said computer system comprising:

a system CPU;

memory;

at least one drive comprising compressed audio data;

a first operating system adapted to control at least said system CPU and said memory;

a second operating system, said second operating system being adapted to cause said system CPU to decompress said compressed audio data and store said decompressed audio data in said memory;

a first switch, the activation of said first switch causing said first operating system to boot; and

a second switch, the activation of said second switch causing said second operating system to boot.

14. A computer system adapted to play audio files, said computer system comprising:

a system CPU;

memory;

a first operating system adapted to control at least said system CPU and said memory;

at least one drive comprising compressed audio data;

a second operating system, said second operating system being adapted to cause said system CPU to decompress said compressed audio data and

store said compressed audio data in said memory; and

a switch, the activation of said switch causing said second operating system to boot.

15. A computer system adapted to play audio files, said computer system comprising:

compressed audio data;

a system CPU;

a first operating system adapted to control at least said system CPU and said memory;

a second operating system, said second operating system being adapted to cause said system CPU to decompress said compressed audio data; and

a switch, the activation of said switch causing said second operating system to boot and cause said system CPU to decompress said compressed audio data.

16. A computer system adapted to play audio files, said computer system comprising:

compressed audio data;

a system CPU;

an audio controller;

a first operating system adapted to control at least said system CPU and said memory;

a second operating system, said second operating system controlling said audio controller and said system CPU, so as to cause said system CPU to decompress said compressed audio data; and

a switch, the activation of said switch causing said second operating system to boot.

17. A computer system adapted to play audio files, said computer system comprising:

compressed audio data;

a system CPU;

an audio controller;

a first operating system adapted to control at least said system CPU;

a second operating system, said second operating system controlling said audio controller and said system CPU, so as to cause said system CPU to decompress said compressed audio data; and

a switch, the activation of said switch causing said second operating system to boot.

18. A computer system adapted to play audio files, said computer system comprising:

a drive comprising at least one audio file;

an audio controller;

an operating system, said operating system being stored in BIOS and adapted to control said audio controller, so as to cause said audio controller to play said at least one audio file; and

a switch, the activation of said switch causing said operating system to boot.

19. A computer system adapted to play audio files, said computer system comprising:

a system CPU;

a drive comprising at least one compressed audio file; and

an operating system, said operating system being stored in BIOS and adapted to control said system CPU, so as to cause said system CPU to decompress said at least one audio file; and

a switch, the activation of said switch causing said operating system to boot.

20. A computer system adapted to play audio files, said computer system comprising:

a system CPU;

memory;

at least one drive comprising compressed audio data; and

an audio controller coupled to said system CPU, memory and drive;

said audio controller being adapted to cause said drive to read said compressed audio data, to cause said system CPU to decompress said compressed audio data, thereby providing decompressed audio data, and to cause said decompressed audio data to be stored in said memory.

21. A computer system as claimed in claim 20, wherein said audio controller is further adapted to place said system CPU in standby state when said system CPU is not decompressing said compressed audio data.

22. A computer system as claimed in claim 20, wherein said audio controller is further adapted to cause said decompressed audio data to be retrieved from said memory for playing.

23. A computer system as claimed in claim 20, wherein said drive is a hard disk, removable disk, floppy disk, magnetic storage medium, optical storage medium, or IDE device.

24. A computer system as claimed in claim 20, wherein said compressed audio data is in MP3, WMA, AAC, or other secured compressed audio format.

25. A computer system as claimed in claim 20, further comprising at least one digital computer bus, wherein said audio controller is coupled to at least one of said system CPU, memory, and drive via said digital computer bus.

26. A computer system as claimed in claim 20, further comprising a mini-OS.

27. A computer system as claimed in claim 20, further comprising an LCD interface for generating signals to an LCD display for displaying song name, file/directory name and/or timing data.

28. A computer system as claimed in claim 20, further comprising a plurality of function keys and a function key interface operable with said plurality of function keys, said function keys generating user commands to said audio controller through said function key interface.

29. A computer system as claimed in claim 28, further comprising a software driver for receiving interrupts generated by at least one of said plurality of function keys and for passing said interrupts to said system CPU.

30. A computer system as claimed in claim 29, further comprising standard

audio player software, wherein said CPU utilizes said interrupts to control said standard audio player software.

31. A computer system as claimed in claim 20, wherein said audio controller is adapted not to cause said drive to read said compressed audio data, nor to cause said system CPU to decompress said compressed audio data , nor to cause said decompressed audio data to be stored in said memory, unless said computer system is off, in hibernate mode, in suspend to HDD mode, or in one of power states S4 or S5.

32. A computer system as claimed in claim 20, wherein said audio controller is adapted not to cause said drive to read said compressed audio data, nor to cause said system CPU to decompress said compressed audio data, nor to cause said decompressed audio data to be stored in said memory, when said computer system is on, in sleep mode, in suspend to RAM mode, or in one of power states S0 or S3.

33. A computer system as claimed in claim 29, wherein said software driver is adapted not to receive said interrupts generated by at least one of said plurality of function keys nor pass said interrupts to said system CPU, unless said computer system is on, in sleep mode, in suspend to RAM mode, or in one of power states S0 or S3.

34. A computer system as claimed in claim 20, wherein said compressed audio data is stored in one or more audio files on said drive, said computer system further comprising a play list software program for creating and storing a play list comprising one or more said audio files.

35. A computer system as claimed in claim 34, wherein said play list software program is executable only when said computer is on or in power state S0.

36. A computer system as claimed in claim 35, wherein said audio controller is further adapted to cause said drive to read said compressed audio data based, at least in part, on said stored play list.

37. A computer system adapted to play audio files, said computer system comprising:

a system CPU;

memory;

at least one drive comprising compressed audio data, said

compressed audio data residing in one or more audio files;

a play list software program for selecting a play list comprising one or more of said audio files; and

an audio controller coupled to said system CPU, memory and drive;

said audio controller being adapted to cause said drive to read at least one said audio file of said play list, to cause said system CPU to decompress the compressed audio data of said file and thereby provide decompressed audio data, and to cause said decompressed audio data to be stored in said memory.

38. A method of playing audio files on a computer system, said method comprising:

reading compressed audio data from the drive of a computer system having at least a drive, a CPU, and a memory;

providing said compressed audio data to said CPU for decompressing said compressed audio data, thereby providing decompressed audio data; and

storing said decompressed audio data in said memory.

39. A method of playing audio files on a computer system as claimed in claim 38, further comprising placing said system CPU in a standby state when said system CPU is not decompressing said compressed audio data.

40. A method of playing audio files on a computer system as claimed in claim 38, further comprising retrieving said decompressed audio data from said memory for playing.

41. A method of playing audio files on a computer system as claimed in claim 38, wherein said drive is a hard disk, removable disk, floppy disk, magnetic storage medium, optical storage medium, flash media, or IDE device.

42. A method of playing audio files on a computer system as claimed in claim 38, wherein said compressed audio data is in MP3, WMA, AAC, or other secured compressed audio format.

43. A method of playing audio files on a computer system as claimed in claim 38, further comprising generating signals to an LCD display for displaying song

name, file/directory name and/or timing data.

44. A method of playing audio files on a computer system as claimed in claim 38, wherein said computer system further comprises a plurality of function keys, and wherein said method further comprises receiving user commands generated by at least one of said plurality of function keys and utilizing said user commands to control said playing.

45. A method of playing audio files on a computer system as claimed in claim 38, further comprising receiving interrupts generated by at least one of said plurality of function keys and passing said interrupts to said system CPU.

46. A method of playing audio files on a computer system as claimed in claim 38, wherein said computer system further comprises standard audio player software, and wherein said method further comprise utilizing said interrupts to control said standard audio player software.

47. A method of playing audio files on a computer system as claimed in claim 38, wherein said steps of reading compressed audio data from the drive of said computer system, providing said compressed audio data to said CPU, and storing said decompressed audio data in said memory, are not performed unless said computer system is off, in hibernate mode, in suspend to HDD mode, or in one of power states S4 or S5.

48. A method of playing audio files on a computer system as claimed in claim 20, wherein said steps of reading compressed audio data from the drive of said computer system, providing said compressed audio data to said CPU for decompressing said compressed audio data into said decompressed audio data, and storing said decompressed audio data in said memory, are not performed when said computer system is on, in sleep mode, in suspend to RAM mode, or in one of power states S0 or S3.

49. A method of playing audio files on a computer system as claimed in claim 45, wherein said steps of receiving interrupts generated by at least one of said plurality of function keys and passing said interrupts to said system CPU are not performed unless said computer system is on, in sleep mode, in suspend to RAM mode, or in one of power states S0 or S3.

50. A method of playing audio files on a computer system as claimed in claim 20, wherein said compressed audio data is stored in one or more audio files on said drive, wherein said method further comprises creating and storing a play list comprising one or more said audio files.

51. A method of playing audio files on a computer system as claimed in claim 50, wherein said step of creating and storing a play list is only performed when said computer is on or in power state S0.

52. A method of playing audio files on a computer system as claimed in claim 51, further comprising reading said compressed audio data from said drive based, at least in part, on said play list.

53. A method of playing audio files on a computer system, said method comprising:

creating and storing a play list comprising a list of compressed audio files residing on one or more drives of a computer system having at least a drive, a CPU, and a memory;

reading said play list;

reading said compressed audio files from said drive based on said play list;

providing said compressed audio data to said CPU for decompressing the data of said compressed audio file into decompressed audio data;

storing said decompressed audio data in said memory; and

retrieving said decompressed audio data from said memory for playing.

54. A method of playing audio files on a computer system, said method comprising:

when said computer system is on, in sleep mode, in suspend to RAM mode, or in one of power states S0 or S3, creating and storing a play list comprising a list of compressed audio files residing on one or more drives of a computer system having at least a drive, a CPU, and a memory; and

when said computer system is off, in hibernate mode, in suspend to HDD mode, or in one of power states S4 or S5, playing the compressed audio files of said play list.

55. A method of playing audio files on a computer system, said method

comprising:

when said computer system is on, in sleep mode, in suspend to RAM mode, or in one of power states S0 or S3, creating and storing a play list comprising a list of compressed audio files residing on one or more drives of a computer system having at least a drive, a CPU, and a memory;

when said computer system is off, in hibernate mode, in suspend to HDD mode, or in one of power states S4 or S5, reading said play list;

when said computer system is off, in hibernate mode, in suspend to HDD mode, or in one of power states S4 or S5, reading said compressed audio files from said drive based on said play list;

when said computer system is off, in hibernate mode, in suspend to HDD mode, or in one of power states S4 or S5, providing said compressed audio data to said CPU for decompressing the data of said compressed audio file into decompressed audio data;

when said computer system is off, in hibernate mode, in suspend to HDD mode, or in one of power states S4 or S5, storing said decompressed audio data in said memory; and

when said computer system is off, in hibernate mode, in suspend to HDD mode, or in one of power states S4 or S5, retrieving said decompressed audio data from said memory for playing.

56. A method of playing audio files on a computer system, said method comprising:

reading compressed audio data from the drive of a computer system, said computer system having at least a drive, a CPU, and a memory;

providing said compressed audio data to said CPU for decompressing said compressed audio data into decompressed audio data;

storing said decompressed audio data in said memory; and

playing said decompressed audio data from said memory.

57. A computer system adapted to play audio files, said computer system comprising:

a system CPU;

memory;

at least one drive comprising compressed audio data, said compressed audio data residing in one or more audio files;

a play list software program for selecting and storing a play list comprising one or more of said audio files; and

an audio controller coupled to said system CPU, memory and drive;

said audio controller being adapted to retrieve said play list and cause said drive to read at least one said audio file of said play list, to cause said system CPU to decompress the compressed audio data of said file and provide decompressed audio data, to cause said decompressed audio data to be stored in said memory, and to cause said decompressed audio data to be played from said memory.

58. A computer system adapted to play audio files, said computer system comprising:

a system CPU;

memory;

at least one drive comprising compressed audio data, said compressed audio data residing in one or more audio files;

a play list software program for selecting and storing a play list comprising one or more of said audio files, said play list containing information related to the location of said audio files including the volume serial number of the drive containing said audio file;

a first operating system adapted to control at least said system CPU and said memory; and

a second operating system, said second operating system being stored in BIOS and adapted to retrieve said play list and cause said drive to read at least one said audio file of said play list, to cause said system CPU to decompress the compressed audio data of said file and provide decompressed audio data, and to cause said decompressed audio data to be stored in said memory.

59. A computer system as claimed in claim 58, further comprising a portable memory media device including a portable memory media reader and a portable memory device, wherein said playlist data being stored on said portable memory

media.

60. A computer system as claimed in claim 59, wherein said portable memory media device selected from the group of Smart Card, PCMCIA, CardBus card and Memory Stick.

61. A computer system adapted to play audio files, said computer system comprising:

a system CPU;

memory;

a first drive comprising compressed audio data, said compressed audio data residing in one or more audio files;

a play list software program for selecting and storing a play list comprising one or more of said audio files, said play list stored on a second drive comprising portable memory media;

a first operating system adapted to control at least said system CPU and said memory; and

a second operating system, said second operating system being stored in BIOS and adapted to retrieve said play list and cause said drive to read at least one said audio file of said play list, to cause said system CPU to decompress the compressed audio data of said file and provide decompressed audio data, and to cause said decompressed audio data to be stored in said memory.

62. A computer system as claimed in claim 61, said play list containing information related to the location of said audio files including volume serial number of the drive containing said audio file.

63. A computer system as claimed in claim 61, said portable memory media device including a portable memory media reader and a portable memory device, wherein said playlist data being stored on said portable memory media.

64. A computer system as claimed in claim 63, wherein said portable memory media device selected from the group of Smart Card, PCMCIA, CardBus card and Memory Stick.

FIGURE 1

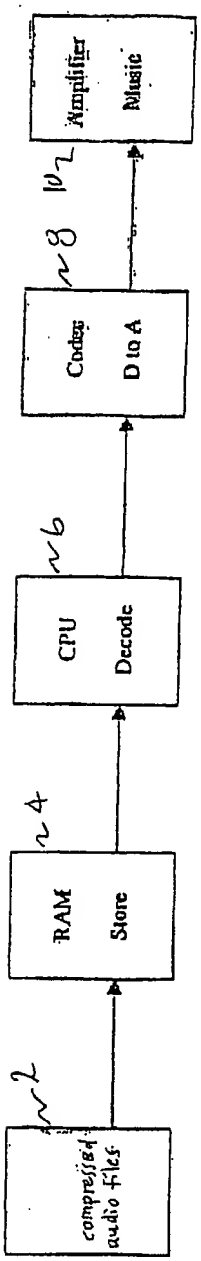

compressed audio files
RAM Store
CPU Decode
Codec D to A
Amplifier Music

FIGURE 2

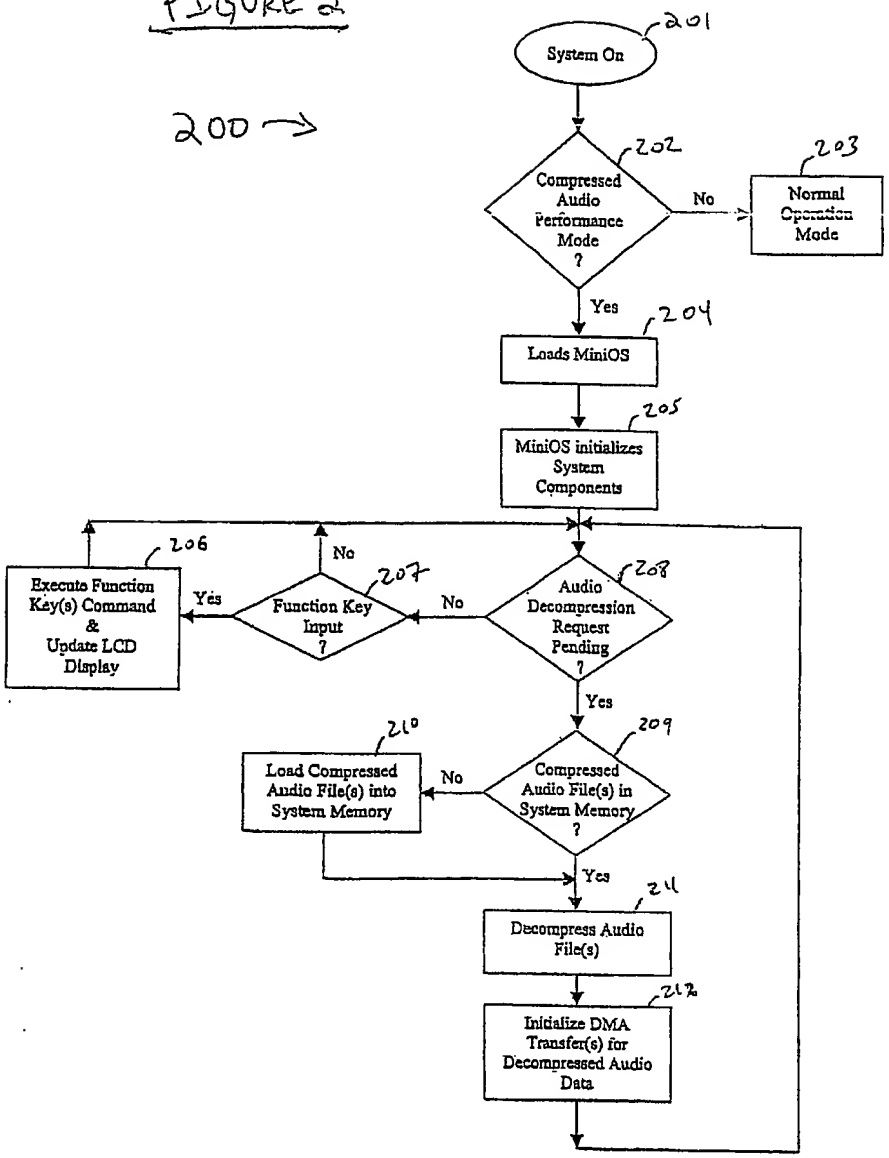
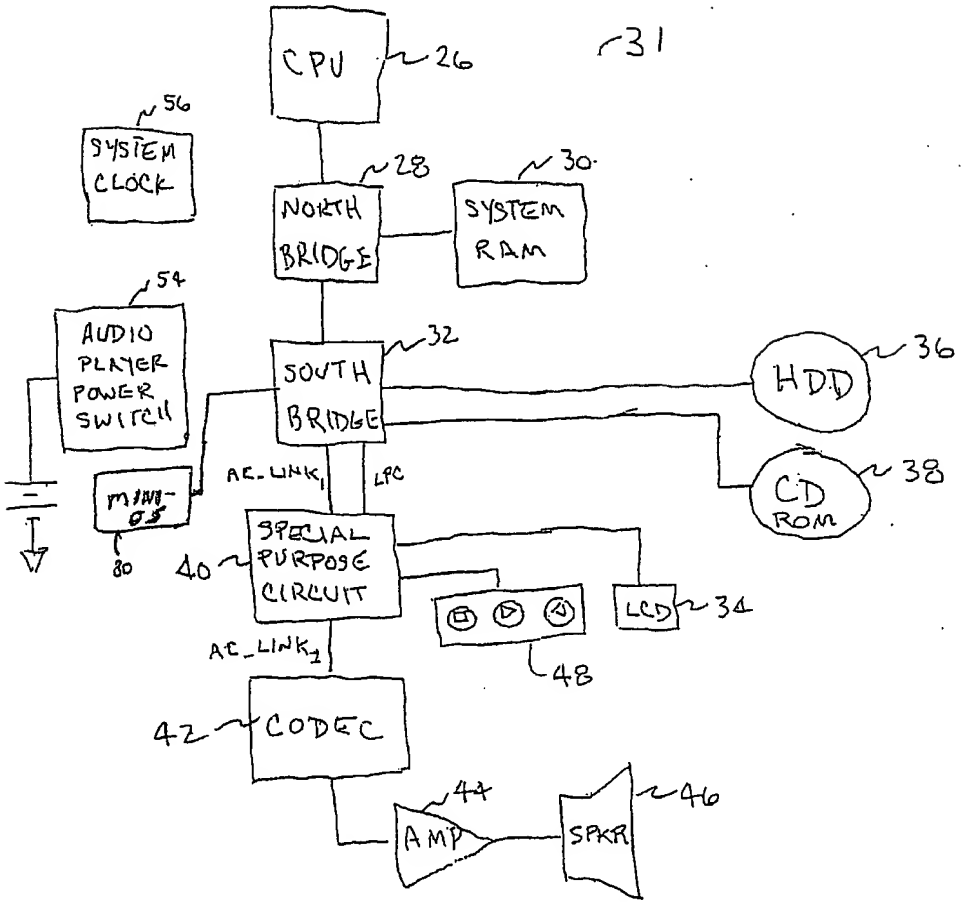

200
201
System On
202
Compressed Audio Performance Mode ?
No
203
Normal Operation Mode
Yes
204
Loads MiniOS
205
MiniOS initializes System Components
206
Execute Function Key(s) Command & Update LCD Display
Yes
207
Function Key Input ?
No
No
208
Audio Decompression Request Pending ?
Yes
210
Load Compressed Audio File(s) into System Memory
No
209
Compressed Audio File(s) in System Memory ?
Yes
211
Decompress Audio File(s)
212
Initialize DMA Transfer(s) for Decompressed Audio Data

FIGURE 3


CPU
26
31
56
SYSTEM CLOCK
28
NORTH BRIDGE
30
SYSTEM RAM
54
AUDIO PLAYER POWER SWITCH
32
SOUTH BRIDGE
HDD
36
CD ROM
38
AC_LINK
LPC
MINI-OS
80
40
SPECIAL PURPOSE CIRCUIT
LCD
34
48
AC_LINK
42
CODEC
44
AMP
SPKR
46

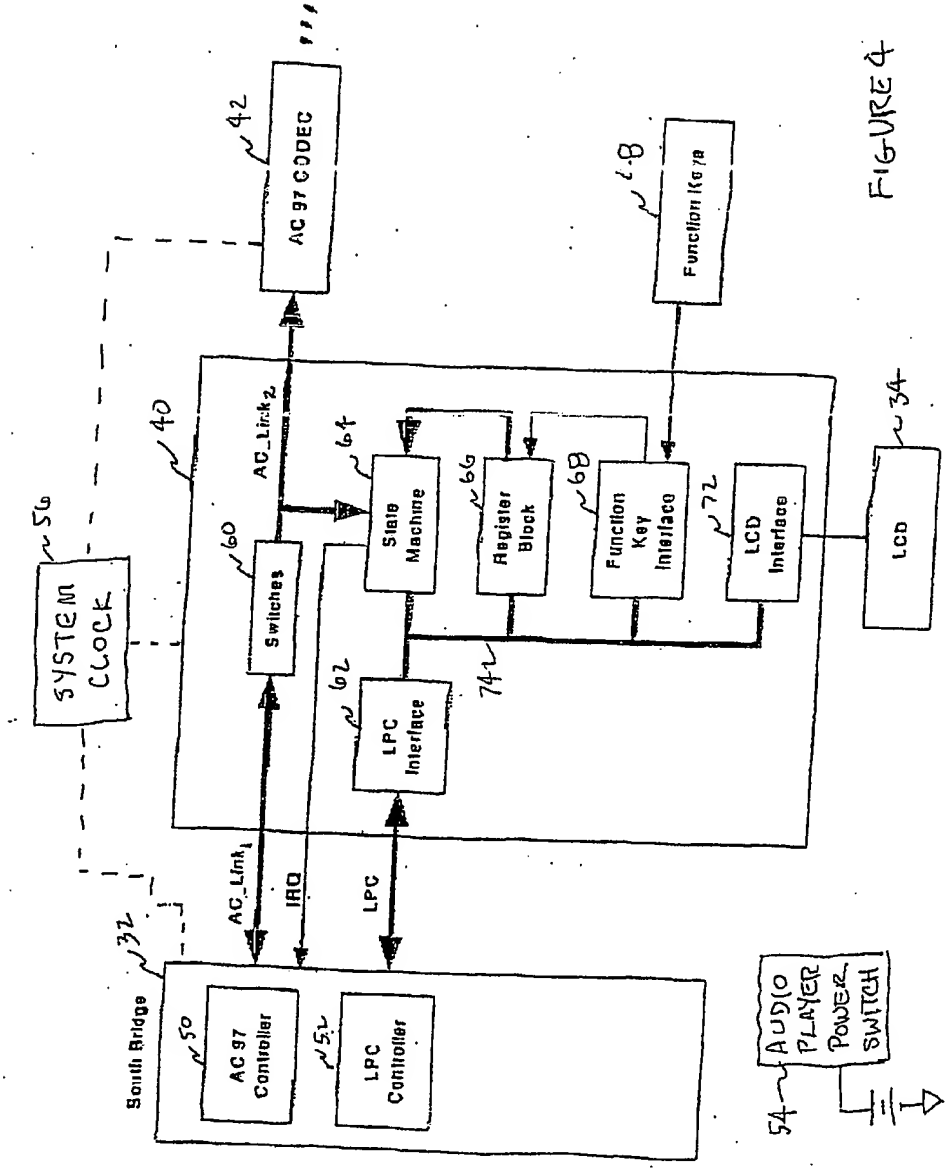
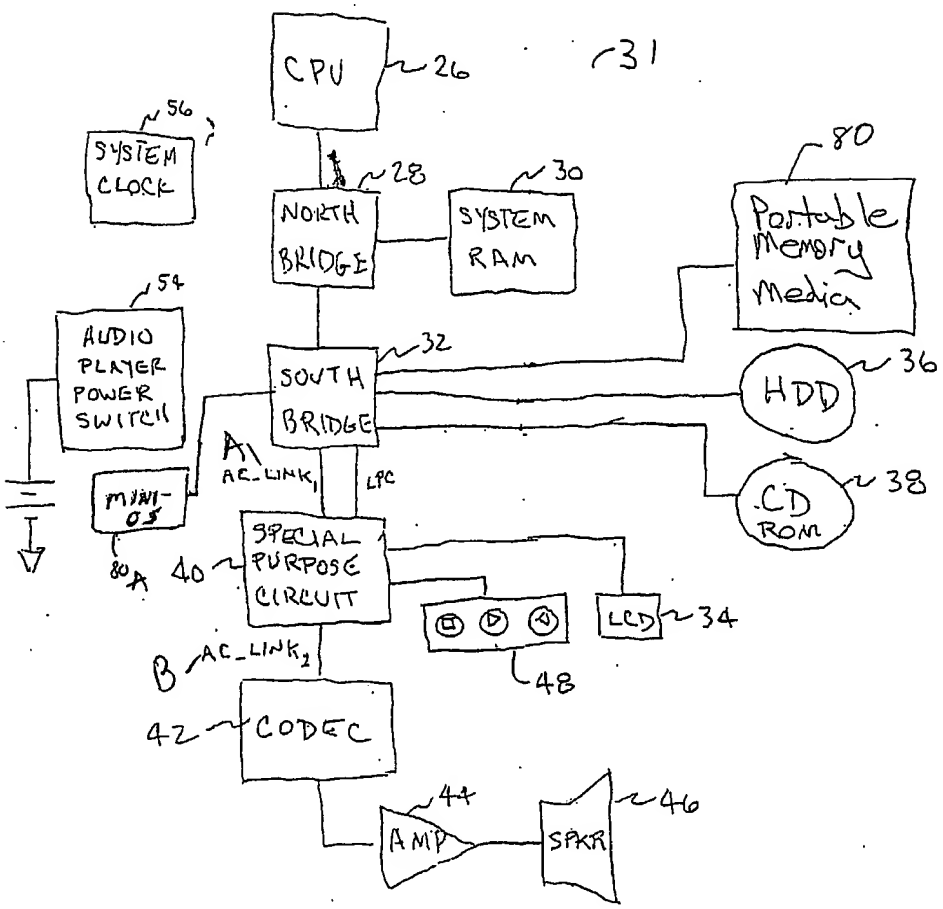

AC 97 CODEC
42
Function Keys
48
40
AC_Link2
64
State Machine
66
Register Block
68
Function Key Interface
72
LCD Interface
LCD
34
56
SYSTEM CLOCK
60
Switches
62
LPC Interface
74
AC_Link1
IRQ
LPC
32
South Bridge
50
AC 97 Controller
52
LPC Controller
54
AUDIO PLAYER POWER SWITCH

FIGURE 4

FIGURE 5


CPU
26
31
56
SYSTEM CLOCK
28
NORTH BRIDGE
30
SYSTEM RAM
80
Portable Memory Media
54
AUDIO PLAYER POWER SWITCH
32
SOUTH BRIDGE
36
HDD
38
CD ROM
A
AC_LINK1
LPC
MINI-OS
80A
40
SPECIAL PURPOSE CIRCUIT
LCD
34
48
B
AC_LINK2
42
CODEC
44
AMP
46
SPKR